

多相位粒度萤火虫同步算法

郝创博 宋萍* 杨诚 武江鹏

(北京理工大学仿生机器人与系统教育部重点实验室 北京 100081)

摘要: 该文针对分布式时间同步算法中,低耦合系数的分组同步问题和大耦合系数的相位振荡失稳现象,提出一种多粒度一致分布式时间同步算法,采用离散多相位粒度模型,将报文交换的相位信息采用多个相位分辨力进行耦合同步,有效地减少了分组现象,加快了同步进程,并提高了算法的执行效率。最后利用仿真实验和传统的 M&S 仿生算法进行对比实验,验证了其在非全连接网络中,可以取得比传统算法更好的同步效果。

关键词: 无线传感器网络; 分布式时间同步; 局部同步问题; 多相位粒度

中图分类号: TP393.0

文献标识码: A

文章编号: 1009-5896(2016)09-2208-07

DOI: 10.11999/JEIT151395

Firefly-inspired Synchronicity Algorithm Based on Multi Granularity Phase

HAO Chuangbo SONG Ping YANG Cheng WU Jiangpeng

(Key Laboratory of Biomimetic Robots and Systems, Beijing Institute of Technology, Beijing 100081, China)

Abstract: Considering that conventional distributed synchronicity algorithm may lead to a state of partial synchronization separately with small couple coefficient or unsteadiness with large couple coefficient, a multi granularity firefly-inspired synchronicity algorithm is proposed. It lets the phase value couple in multi granularity by its divergence in time and phase, which can relieve the issue of partial synchronization and speed up the convergence process. Its performance is tested by simulation in a non-fully connect network by comparing with the conventional M&S algorithm. The result shows that it works better.

Key words: Wireless Sensor Network (WSN); Distributed synchronization; Partial synchronization; Multi granularity

1 引言

时间同步在无线传感器网络中得到广泛应用。它为无线传感器网络提供了统一的时间基准,对无线测试数据时间相关性建立以及分时休眠等技术具有至关重要的作用。其作为无线传感器网络中研究热点之一,一直备受关注。在早期研究中,时间同步仅仅是主从关系之间的时间信息交换^[1-3]。这种分层形式的同步算法可在简单的星形拓扑结构中取得良好的同步效果,但在复杂的网络中常常具有可扩展性差,鲁棒性差,同步误差随路由跳数增加而累积,同步路由建立复杂等缺点。

随着时间同步技术的发展,人们从大自然中找到灵感,发明了一类分布式同步算法^[4,5]。典型的模型和算法如文献[6]针对心肌细胞的分布式同步跳动

建立了一种脉冲耦合模型;文献[7]在其基础上针对萤火虫的分布式同步闪烁建立了 M&S 脉冲耦合动态学模型。文献[8]等利用该模型发展为萤火虫同步算法并运用于无线传感器网络时间同步。基于此类分布式时间同步算法的节点在网络中的地位是对等的,它们利用其邻居节点的时间或相位信息进行调整进而到达同步的目的。该同步机制和路由机制相互独立,避免了主从关系算法的上述缺点。因此该算法具有抗毁伤性和鲁棒性强,易扩展,多跳累积误差小等特点,适合于大规模无线传感器网络。文献[9]总结了分布式同步典型算法,并分析了延时,漂移,干扰等因素对其收敛的影响。最近,优化研究^[10-16]多围绕分布式时间同步算法的稳定性,收敛速度和通信效率等方面开展。如文献[10-12]使用特定的事件触发来修正邻居节点的相位信息,采用时间离散模型使用离散的数据包修正连续的本地时钟,但没有考虑延时与算法复杂度等实际问题。文献[13-15]针对现实中的噪声影响,提出具有稳定性和一定精度的同步算法,但是没有避免非全连接中

收稿日期: 2015-12-09; 改回日期: 2016-05-04; 网络出版: 2016-07-04

*通信作者: 宋萍 sping2002@bit.edu.cn

基金项目: 国家自然科学基金(61202433)

Foundation Item: The National Natural Science Foundation of China (61202433)

的分组现象对同步进程的阻碍。文献[16]研究了非全连接网络中,影响同步稳定性和速度的因素,并提出了一种通过改变邻居节点数目来加速收敛的同步策略,仅仅只做了理想模型(无耦合延时)下的研究。目前这些研究仅针对时间离散相位连续的同步方式。时间离散相位连续的同步方式可以在一定程度上解决同步传输占用带宽的问题,但会引入分组现象(局部同步现象)和标准时钟漂移等问题^[17]。目前针对时间和相位均离散的同步算法研究鲜有涉及。

针对连续相位模型的上述问题,本文提出一种多相位粒度萤火虫同步算法,有别于文献[10-16]中的离散时间连续相位模型,采用时间和相位离散的多相位空间耦合算法,在建模和算法设计中考虑延时、网络连通性、算法复杂度实际问题,减少同步过程中的局部同步的半稳定状态持续时间,有利于加速全局同步,解决分组现象,并且更适合在低性能MCU中运行,提高了算法执行效率。

2 预备知识

2.1 CHARLES 模型

CHARLES 针对心肌细胞的同步跳动进行分析,将每个心肌细胞看作为一个电路中的RC振荡器,并将其自由状态和耦合状态分别建立数学模型^[6]。

在自由状态时,振荡器以如下速度进行充电:

$$\frac{dU}{dt} = C - kU, \quad 0 \leq U \leq 1 \quad (1)$$

式中, U 代表归一化后的振荡器电压, C 代表理想的充电速度, k 表示电阻漏电流因子。振荡器自由振荡过程如下:节点以由一个固定的电流对电容进行充电,引起电容电压的上升。随着电容电压的上升,由于电阻的漏电效应增强,导致电容电压的上升速率逐渐变慢。当归一化后的电容电压达到阈值1时,电容迅速放电,电压很快地降为0。

在耦合状态下,当一个振荡器的电容放电时,会与其他阻容振荡器之间电耦合,从而把其他振荡器的电容电压提升一个很小的增量 ε , 而正是这种耦合的作用,使得振荡器的电容电压趋于相同,最终达到同时放电状态。即

$$U_i(t) = 1 \Rightarrow \begin{cases} U_j(t^+) = \min(1, U_j(t) + \varepsilon), \forall j \neq i \\ U_i(t^+) = 0 \end{cases} \quad (2)$$

2.2 M&S 萤火虫模型及同步算法

MIROLLO 与 STROGATZ 参考 CHARLES 模型,对萤火虫的同步闪烁建模和同步算法研究,其与 CHARLES 模型的机理相同均采用振荡器脉冲耦合(Pulse Coupling Oscillator, PCO)正反馈,并引入相位 ϕ 作为中间变量,建立振荡器动力学模型

(MIROLLO & STROGATZ 模型, M&S 模型):

$$U = f(\phi) \quad (3)$$

式中, f 是一个光滑单调凹函数,并且过点(0,0)和(1,1)。在 M&S 算法中,振荡器状态 U 的变化被映射到相位 ϕ 的变化,设节点的充电周期为 T ,式(1)和式(2)被等效为

$$d\phi/dt = 1/T \quad (4)$$

$$\phi_i(t) = 1 \Rightarrow \forall j \neq i,$$

$$\phi_j(t) = \begin{cases} f^{-1}(f(\phi_j(t)) + \varepsilon), & f(\phi_j(t)) + \varepsilon < 1 \\ 0, & f(\phi_j(t)) + \varepsilon \geq 1 \end{cases} \quad (5)$$

即节点在自由状态时相位以固定速率 $1/T$ 连续增长;在耦合状态时,若在 t 时刻收到触发信号则将相位增长至 $f^{-1}(f(\phi_j(t)) + \varepsilon)$,若该值大于阈值1,则节点相位立刻归零并触发。

最终,MIROLLO 与 STROGATZ 证明了在无延时全连接网络下该耦合方式可以使任意初始状态的网络同步。

3 多相位粒度建模

萤火虫同步数学模型 M&S 中的相位模型是连续的,适合于在模拟电路中通过硬件的方式实现同步,而在数字电路中通过逻辑判断和计算实现萤火虫分布式同步,需要将 M&S 模型中的相位离散化,从而简化同步过程在 MCU 中的处理。本文所述相位粒度主要是指相位离散过程中量化的分辨率。由式(5)可知相位的触发阈值为1,我们定义相位粒度为

$$\Delta\phi = 1/n_{\max} \quad (6)$$

其中, $n_{\max} \in Z^+$, 表示离散相位的量化级数。使用此相位粒度所表示的节点相位所有可能值的集合为

$$S = \{\Delta\phi, 2\Delta\phi, \dots, n_{\max}\Delta\phi\} \quad (7)$$

我们称 S 为该相位粒度的相位空间。当前的离散相位值和真实相位值之间满足

$$n(t)\Delta\phi \leq \phi < [n(t) + 1]\Delta\phi \quad (8)$$

式中, $n(t)$ 表示当前时刻的离散相位值。真实相位在该分布范围内呈均匀分布,根据其分布范围,则真实相位值的期望和方差为

$$E(\phi) = [n(t) + 0.5]\Delta\phi \quad (9)$$

$$D(\phi) = \Delta\phi^2/12 \quad (10)$$

由此可见,相位粒度越小,真实相位值的分布范围越小,精度越高,反之分布范围越大,精度就越低。较大的粒度导致量化级数较少,状态空间的维数较小,有利于离散相位值的同步,但是会影响

真实相位值的精度,反之,状态空间的维数较大,不利于离散相位值的收敛。因此,本节提出一种多相位粒度模型,通过减少相位空间维数加快同步进程,减小分组的可能,同时提供真实相位值精度。

在多相位粒度模型中,设将相位的离散的量化过程分 m 级,每一级使用不同的相位粒度对相位进行离散,形成离散的多相位空间。因此对应每个量化级数的相位空间,相位离散后的离散相位阈值也分 m 级,并且满足:

$$n_{1\max}\Delta\phi_1 = n_{2\max}\Delta\phi_2 = \dots = n_{m\max}\Delta\phi_m = 1 \quad (11)$$

其中, $n_{l\max}$ 表示第 l 级的离散相位阈值,即该级量化级数。

为了简化模型,方便在各个相位空间中进行同步,设每一级相位粒度是下级的 n_{\max} 倍。故有

$$n_{\max}\Delta\phi_1 = n_{\max}^2\Delta\phi_2 = \dots = n_{\max}^m\Delta\phi_m = 1 \quad (12)$$

$$\Delta\phi_l = \frac{1}{n_{\max}^l}, \quad l \in \{0 < l \leq m \mid l \in Z^+\} \quad (13)$$

在多相位粒度模型中,某一时刻 t 真实的相位值可以由多个相位空间的离散相位值表示,将它们以矩阵的形式结合起来如式(14):

$$\mathbf{N}(t) = \begin{bmatrix} n_1(t) \\ n_2(t) \\ \vdots \\ n_m(t) \end{bmatrix} \quad (14)$$

我们将矩阵 \mathbf{N} 定义为节点的多粒度状态矩阵。在多相位粒度下的自由状态模型与M&S的自由状态模型类似,但是将充电过程分布在各级粒度的相位空间中。由于各级的相位粒度不同,因此各级相位空间中提升单位相位粒度所需的充电时间也是不同的。设节点的充电周期为 T ,则第 l 级中增加单位相位粒度所需的时间为

$$\Delta t_l = \Delta\phi_l T = T/n_{\max}^l \quad (15)$$

在耦合状态下,由于相位的离散化,相位增长不连续,并且其增量必须为相位粒度的整数倍,为了保证式(3)中相位映射函数的凹函数特性,在多相位粒度模型中,各级相位粒度空间在耦合的向量增量为一个阶梯递增函数。在本文中,我们使用一个阶跃函数进行简化。即

$$n_l(t^+) - n_l(t) = \begin{cases} 1, & n_l(t) \geq \tau \\ 0, & n_l(t) < \tau \end{cases}, \quad l = 1, 2, \dots, m \quad (16)$$

故多相位粒度模型在耦合状态下的动态方程如式(17):

$$\begin{aligned} n_{ij} &\geq n_{l\max} \Rightarrow \forall j \neq i, \\ n_l(t^+) &= \begin{cases} n_l(t)+1, & n_l(t) \geq \tau \\ n_l(t), & n_l(t) < \tau \end{cases} \end{aligned} \quad (17)$$

4 多相位粒度萤火虫同步算法

第3节中介绍了节点的数学模型,本节提出一种多相位粒度萤火虫同步算法(下文简称多粒度同步算法),利用多相位粒度数学模型对任意初始相位节点集进行同步,并对算法进行稳定性分析。

整个同步算法主要分为3个任务:离散相位值自增长,节点发送同步报文,同步报文的处理。

4.1 离散相位值的自增长

节点在自由状态下各级相位空间中的离散相位值按照式(15)的速度增长。不同于一般连续相位模型的是,多相位粒度的相位空间属于离散空间,其相位的增长属于阶梯状增长,如图1所示为量化等级为4,相位周期为1s的前3级相位空间相位的自增长情况。根据式(15)可知,需要每隔 Δt_l 将节点第 l 级的离散相位值加1。由于采用离散时间的相位自增长方式,不需要节点对相位信息进行实时更新,方便使用定时器实现,节约节点处理资源。

4.2 同步报文发送

模型采用离散时间模型的事件耦合方式,不需要节点之间连续的交换相位信息,减少了通信负荷。如RC振荡器的脉冲耦合相同,由式(11)和式(15)可知,在多粒度的相位模型中,只要任意一级的离散相位值达到其对应的量化级数,则其余各级的相位值均达到对应的量化级数,此时节点触发(相位达到阈值),并向其邻居节点发送同步报文。由于各级为同时达到阈值,故同步报文中暗含了各尺度相位空间中的相位信息,以便在各个相位尺度空间进行相位耦合。

需要注意的是,当网络接近同步时,节点在很多的时间内需要广播大量同步报文,容易造成网络拥堵。因此在节点触发后,在应用层上为每个同步报文设置一个随机长度的避让时间,并将该避让时间 δ_{app} 写入同步报文中。由于无线网络通信中,节点发出在MAC层上普遍采用CSMA/CA的冲突避免机制,为了可以准确获取节点的触发时刻信息,需要将CSMA/CA所产生的避让时间 δ_{ca} 同样写入同步报文中。

4.3 同步报文处理

节点在同步过程中接收邻居节点发送的同步报文,并在各个空间进行耦合同步处理是整个同步算法的核心。为方便说明,我们假设节点 j 收到了节点 i 发送的同步报文,则处理过程如下:

(1)节点 j 记录节点 i 的同步报文到达时刻的所

有离散相位值: n_1, n_2, \dots, n_m ;

(2)解析同步报文中各个避让时间 δ_{app} 和 δ_{ca} ;

(3)计算出节点 i 触发时刻, 节点 j 的真实离散相位值。

为了方便表示和计算, 将节点的真实相位值使用多相位粒度表示, 根据式(13)可知:

$$\phi = n_1^* \Delta \phi_1 + n_2^* \Delta \phi_2 + \dots + n_m^* \Delta \phi_m \quad (18)$$

$$n_l^* = \text{mod}(n_l, n_{\max}) \quad (19)$$

式(19)中, mod 函数为取模函数。将避让时间转换到以最小相位粒度(第 m 级)衡量的离散相位值。

$$n_{dy} = (\delta_{app} + \delta_{ca}) / \Delta \phi_m \quad (20)$$

故节点 i 触发时刻, 节点 j 的相位第 m 级的离散相位值应减去 n_{dy} , 若 $n_m^* < n_{dy}$, 则向前一级离散相位值借位, 然后将 $n_m^* + n_{\max} - n_{dy}$ 作为节点 j 的相位, 如果不够继续借位, 各级以此类推;

(4)对各级离散相位值进行耦合。在第(3)步中通过式(19), 将每个相位粒度对应的相位值变为量化级数同为 n_{\max} , 真实相位周期不同的相位空间。将各级离散相位值 n_l^* 按照式(17)所规定的耦合动态方程进行耦合, 并将计算出的各级状态空间的相位值调整量应用于原节点相位矩阵中, 过程如下:

$$n_l^*(t^+) - n_l^*(t) = \begin{cases} 1, & n_l^*(t) \geq \tau \\ 0, & n_l^*(t) < \tau \end{cases}, \quad l = 1, 2, \dots, m \quad (21)$$

$$\Delta \mathbf{N} = \begin{bmatrix} n_1^*(t^+) - n_1^*(t) \\ n_2^*(t^+) - n_2^*(t) \\ \vdots \\ n_m^*(t^+) - n_m^*(t) \end{bmatrix} \quad (22)$$

$$\mathbf{N}(t^+) = \Delta \mathbf{N} + \mathbf{N}(t) \quad (23)$$

4.4 同步稳定性分析

类似于文献[16]中的稳定性证明, 由第3节建模过程可知, 该算法模型将其映射至多维马氏链, 算法的运行过程相当于马氏链的状态转移过程。马氏

链中的各个状态空间用集合 $S = \{s_1, s_2, \dots, s_{n^*}\}$ 表示, 该状态空间中的每个元素 s 由各节点状态的多粒度状态矩阵 \mathbf{N} 组成。设 \mathbf{P} 表示马尔可夫链的转移概率矩阵。当网络强连通且节点数有限时, 状态空间 S 为有限状态空间, 算法迭代映射到马氏链是有限不可约遍历链。设 $\boldsymbol{\pi}$ 为该马氏链中的唯一稳态分布。根据马氏链的相关理论可知, 存在转移次数 k , 使 $\mathbf{P}^k > 0$, 且 $\lim_{m \rightarrow \infty} \mathbf{P}^m = \mathbf{1}\boldsymbol{\pi}^T$ 。由式(24)可知^[16], 稳态为马氏链吸收态, 随着算法迭代, 使其他节点的多粒度状态矩阵 \mathbf{N} 趋于同步, 最终达到稳定。

$$\lim_{m \rightarrow \infty} t^{(m)} = \lim_{m \rightarrow \infty} \mathbf{P}^m t^{(0)} = \mathbf{1} \mathbf{s}^T \boldsymbol{\pi}^T t^{(0)} \\ \Rightarrow \forall s_i, \quad \lim_{m \rightarrow \infty} t_i^{(m)} = \sum_{k=1}^n \boldsymbol{\pi}_k t_k^{(0)} \quad (24)$$

而根据马氏链收敛条件可知, 该算法适用于有限节点数目的强连接网络, 即对于网络中的任意两个节点可以通过直接或多跳的方式进行消息传递。

5 仿真实证

本节中将上文介绍的多粒度同步算法进行仿真, 为了说明该算法在稳定性和收敛速度方面所做的贡献, 文中同时对传统的 M&S PCO 萤火虫同步模型(下文简称 M&S 算法)进行仿真作为对比, 并以 MCU 为平台, 模拟执行两种算法, 对比其执行时间, 确定算法的执行效率。

5.1 仿真条件

文献[7]中对 M&S 模型的论述基于全连接网络条件, 即网络中每个节点可以与其他任意节点直接通信, 但这往往与现实不符, 因此本次仿真主要采用非全连接网络进行仿真实验, 并于 5.2.2 节中增加多相位粒度算法在全连接网络中应用的同步速度与精度说明。仿真实验中的一些通用参数规定如下: 整个网络由 20 个节点组成, 随机分布在 $1000 \text{ m} \times 1000 \text{ m}$ 的区域中, 节点的通信距离为 450 m, 节点所成无方向网络图如图 2(a)所示。节点的同步周

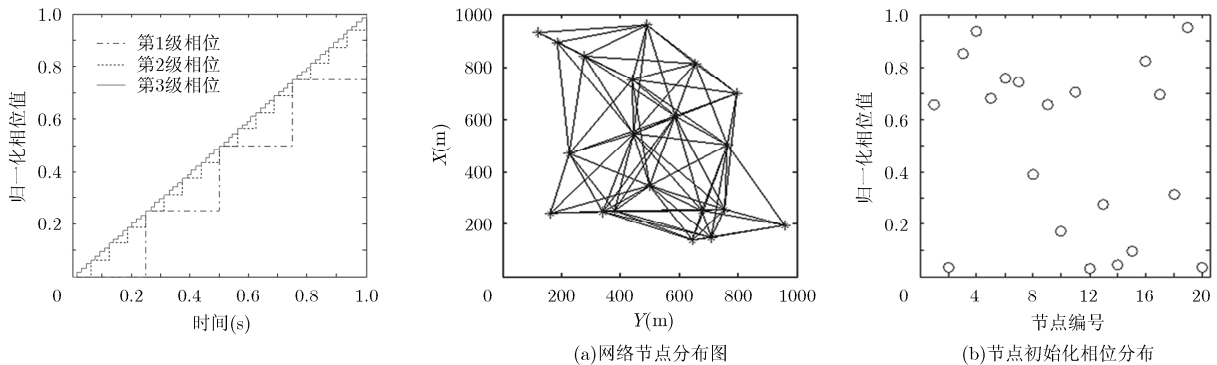


图 1 各相位空间中的相位自增长

图 2 节点分布网络及初始化相位

期为 1 s, 初始相位如图 2(b)所示。从文献[9]可知, 时钟漂移和传输延时在分布式网络中对同步的影响是等效的。为了模拟实现中的时钟漂移和延时, 我们为节点的相位增长速度设置了 1%的误差(即每秒累积归一化相位误差为 ± 0.005)。仿真的时间粒度为 $0.01 \mu\text{s}$, 仿真时长为 100 s。

在传统模型中, 我们采用 M&S 算法, 使用文献[7]中的相位模型对网络进行同步, 仿真实验中采用大小两组耦合系数, 分别为 0.001 和 0.01; 在多粒度同步算法中, 节点的相位粒度分为 3 级, 量化级数分别为: 50, 2500, 125000。

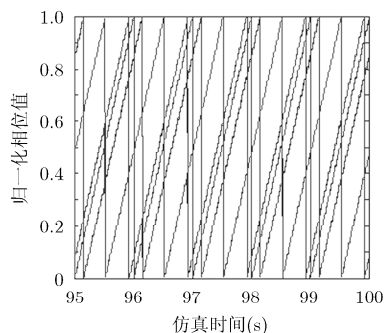
5.2 仿真结果

5.2.1 分组效果 图 3 和图 4 表示同步过程稳定后(仿真时间的最后 15 s)的节点相位细节。从图 3 (a)中可看出耦合系数 0.001 的传统算法收敛后, 全部节点大体分为 3 组, 3 组之间的相位差有轻微抖动。当耦合系数增大至 0.01 后, 从图 3(b)中可以明显看出分组的组数减少, 但是每一组中相位误差变大相位差的抖动较剧烈。由此可知在传统的 M&S 算法在耦合系数的大小决定了同步效果, 当耦合系数较大时, 同步分组现象得到缓解, 但同步精度较差相位抖动明显。而耦合系数较小时, 网络呈现局部收敛, 分组现象严重。图 4 为使用多粒度同步算法的

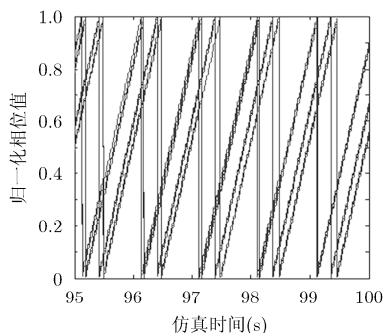
收敛后的各节点相位。可见该算法可以很好地解决分组现象并且相位同步精度较高, 各节点的相位抖动不严重, 整体取得了较好的同步效果。

5.2.2 同步精度和速度 图 5(a)表示网络中任意两个节点同步相位误差的最大值, 可看出无论采取何种耦合系数, 传统的 M&S 算法的同步精度均不及多粒度同步算法。这是由于在非全连接网络中, M&S 算法的分组同步现象导致。采用多粒度同步算法的仿真实验, 网络的相位稳定后, 归一化相位误差维持在 0.025 以下, 并没有受 1%相位增长速度误差的影响, 达到了一定的同步效果。图 5(b)所示为同等实验条件下去除相位增长速度误差后或使用全连接网络仿真理想情况的同步结果, 其同步误差均小于图 5(a)。对比图 5(a)和图 5(b), 得出图 5(a)所示多粒度同步算法收敛后的同步相位误差是由每个节点的相位增长速度误差引起, 而增加网络的连通性可以加速同步并在一定程度上减小这种误差但不能根除。

从图 5(a)中可以看出, 多粒度同步算法的收敛速度与大耦合系数的 M&S 算法收敛相近, 明显快于小耦合系数的 M&S 算法。故多粒度同步算法有一定的加速收敛作用。



(a)小耦合系数M&S算法同步分组情况



(b)大耦合系数M&S算法同步分组情况

图 3 M&S 算法同步分组情况

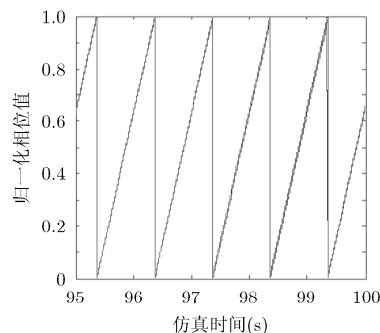
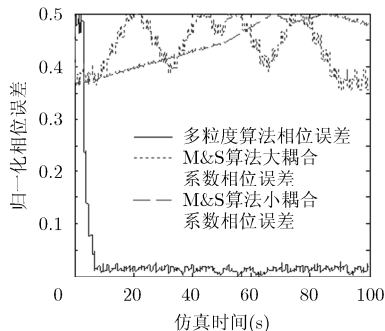
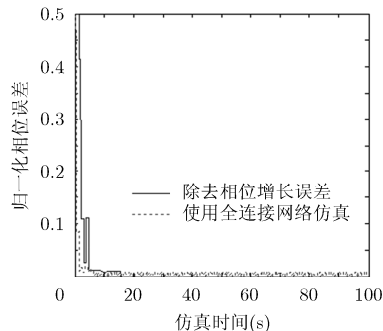


图 4 多粒度同步算法同步分组情况



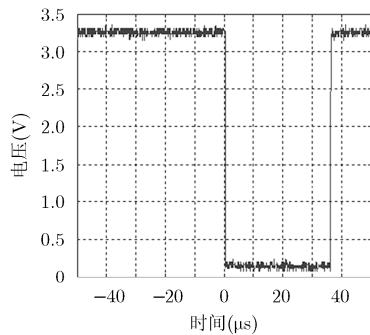
(a)各算法相位误差随时间变化情况



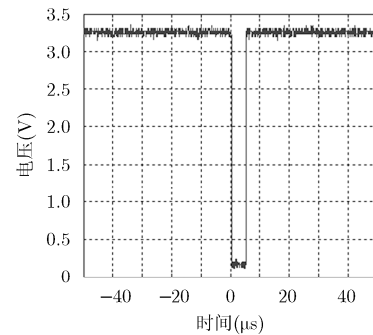
(b)理想状态下多粒度同步算法相位误差随时间变化情况

图 5 同步精度图

5.2.3 算法执行效率 由于无线传感器网络中常常使用运算能力有限的低成本低功耗 MCU, 故算法的时间复杂度是决定算法使用性的关键。为此, 我们利用 ATMEL 公司的 ATMEGA256 芯片模拟执行触发程序 100 次, 通过 IO 口的低电平时间表示触发处理程序执行的时间。实验结果如图 6, 运行 100 次触发算法, M&S 算法耗时约为 $36.2 \mu\text{s}$ (如图 6(a)), 多粒度同步算法耗时约为 $5.2 \mu\text{s}$ (如图 6(b))。从图 6(a)和图 6(b)对比中可以明显看出多粒度同步算法具有更高的执行效率。



(a) 表征M&S算法运算速度的IO电平波形



(b) 表征多粒度同步算法运算速度的IO电平波形

图 6 表征各算法运算速度的 IO 电平波形

6 结束语

分布式时间同步技术具有可扩展性强, 鲁棒性强等特点, 是无线传感器网络中的关键技术。本文提出了一种多相位粒度萤火虫时间同步算法, 并通过仿真实验验证。结果表明该算法有效解决了传统分布式时间同步中的分组现象和同步速度较慢的问题, 并且提高了同步算法的执行效率。我们将在后续工作中进一步解决分布式算法中的通信拥堵问题, 并进一步提高能量效率, 使之成为更加实用的分布式同步算法。

参考文献

- [1] ELSON J, GIROD L, and ESTRIN D. Fine-grained network time synchronization using reference broadcasts[C]. The 2002 Usenix Symposium on Operating Systems Design and Implementation, Berkeley, CA, USA, 2002: 9-11.
- [2] GANERIWAL S, KUMAR R, and SRIVASTAVA M B. Timing-sync protocol for sensor networks[C]. SenSys'03: the First International Conference on Embedded Networked Sensor Systems, Los Angeles, CA, USA, 2003: 138-149.
- [3] XU N, ZHANG X, WANG Q, et al. An improved flooding time synchronization protocol for industrial wireless networks[C]. The 2009 International Conference on Embedded Software and Systems, Hangzhou, China, 2009: 524-529. doi: 10.1109/ICISS.2009.10.
- [4] BUSCH N E, VINNICHE.N K, WATERMAN A T, et al. Waves and turbulence[J]. *Radio Science*, 1969, 4(12): 1377. doi: 10.1029/RS004i012p01377.
- [5] HOLDEN A V. From clocks to chaos - the rhythms of life - glass, L, mackey, MC[J]. *Nature*, 1988, 336(6195): 119. doi: 10.1038/336119a0.
- [6] PESKIN C S and Courant Institute of Mathematical Sciences. *Mathematical Aspects of Heart Physiology*[M]. New York, USA, New York: Courant Institute of Mathematical Sciences, 1975: 278.
- [7] MIROLLO R E and STROGATZ S H. Synchronization of pulse-coupled biological oscillators[J]. *SIAM Journal on Applied Mathematics*, 1990, 50(6): 1645-1662. doi: 10.1137/0150098.
- [8] HONG Y W and SCAGLIONE A. A scalable synchronization protocol for large scale sensor networks and its applications[J]. *IEEE Journal on Selected Areas in Communications*, 2005, 23(5): 1085-1099.
- [9] SIMEONE O, SPAGNOLINI U, BAR-NESS Y, et al. Distributed synchronization in wireless networks[J]. *IEEE Signal Processing Magazine*, 2008, 25(5): 81-97. doi: 10.1109/MSP.2008.926661.
- [10] LIU T, CAO M, and HILL D J. Distributed event-triggered control for output synchronization of dynamical networks with non-identical nodes[C]. The 2014 53rd IEEE Annual Conference on Decision and Control, Los Angeles, CA, USA, 2014: 3554-3559. doi: 10.1109/CDC.2014.7039941.
- [11] KADOWAKI Y and ISHII H. Event-based distributed clock synchronization for wireless sensor networks[J]. *IEEE Transactions on Automatic Control*, 2015, 60(8): 2266-2271.
- [12] SUN W L, STROM E G, BRANNSTROM F, et al. Random

- broadcast based distributed consensus clock synchronization for mobile networks[J]. *IEEE Transactions on Wireless Communications*, 2015, 14(6): 3378–3389.
- [13] HE J, DUAN X, CHENG P, *et al.* Distributed time synchronization under bounded noise in wireless sensor networks[C]. The 2014 53rd IEEE Annual Conference on Decision and Control, Los Angeles, CA, USA, 2014: 6883–6888. doi: 10.1109/CDC.2014.7040470.
- [14] TANG Y, GAO H, LU J, *et al.* Pinning distributed synchronization of stochastic dynamical networks: A mixed optimization approach[J]. *IEEE Transactions on Neural Networks and Learning Systems*, 2014, 25(10): 1804–1815. doi: 10.1109/TNNLS.2013.2295966.
- [15] XIAO H, ISSHIKI T, LI D, *et al.* Distributed synchronization for message-passing based embedded multiprocessors[C]. The 25th IEEE International Conference on Application-Specific Systems, Architectures and Processors, Zurich, Switzerland, 2014: 82–83. doi: 10.1109/ASAP.2014.6868640.
- [16] 李立, 刘勇攀, 杨华中, 等. 无线传感器网络分布式一致时间同步协议的收敛分析及加速设计[J]. 电子与信息学报, 2010, 32(9): 2045–2051. doi: 10.3724/SP.J.1146.2009.01234.
- LI Li, LIU Yongpan, YANG Huazhong, *et al.* Convergence analysis and accelerating design for distributed consensus time synchronization protocol in wireless sensor networks[J]. *Journal of Electronics & Information Technology*, 2010, 32(9): 2045–2051. doi: 10.3724/SP.J.1146.2009.01234.
- [17] KURAMOTO Y. *Chemical Oscillations, Waves, and Turbulence* [M]. Berlin, New York: Springer-Verlag, 1984: 156.
- 郝创博: 男, 1989 年生, 博士生, 研究方向为无线传感器网络.
宋萍: 女, 1972 年生, 教授, 研究方向为传感与机电控制.
杨诚: 男, 1988 年生, 博士生, 研究方向为测控技术.
武江鹏: 男, 1987 年生, 博士生, 研究方向为无线传感器网络.